

CS5101: Design and Analysis of Algorithms



Arijit Mondal

Dept of CSE

`arijit@iitp.ac.in`

`https://www.iitp.ac.in/~arijit/`

General Information

- Class timings
 - Monday — 1200-1300, LT101
 - Monday — 1400-1500, LT101
 - Tuesday — 1200-1300, LT101
 - Thursday — 1400-1500, R303

Books

- Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, *Introduction to Algorithms*, Third Edition, MIT Press/McGraw-Hill
- Sanjoy Dasgupta, Christos H. Papadimitriou and Umesh V. Vazirani, *Algorithms*, Tata McGraw-Hill, 2008.
- Steven Skiena, *The Algorithm Design Manual*, Springer
- Jon Kleinberg and Éva Tardos, *Algorithm Design*, Pearson, 2005.
- Robert Sedgewick and Kevin Wayne, *Algorithms*, fourth edition, Addison Wesley, 2011.
- Udi Manber, *Algorithms – A Creative Approach*, Addison-Wesley, 1989
- Jeff Erickson, *Algorithms*
- Tim Roughgarden, *Algorithms Illuminated*

Evaluation

- Tentative break-up
- Two quizzes - 20%
- Midsem - 30%
- Endsem - 50%

CS5101: Design and Analysis of Algorithms

Introduction



Arijit Mondal

Dept of CSE

arijit@iitp.ac.in

<https://www.iitp.ac.in/~arijit/>

Algorithm

- Is it a jumbled form of *logarithm*?

Algorithm

- Is it a jumbled form of *logarithm*?
- The word algorithm came into existence sometime after 1957
- Closest word that existed was **algorism** – it means the process of doing arithmetic using Arabic numerals

Algorithm

- Is it a jumbled form of *logarithm*?
- The word algorithm came into existence sometime after 1957
- Closest word that existed was **algorism** – it means the process of doing arithmetic using Arabic numerals
- It came from the name of a famous Persian textbook author, Abu Abd Allah Muhammad ibn Musa al-Khwarizmi
- He wrote the celebrated Arabic text Kitab al-jabr wa'l-muqabala ("Rules of restoring and equating")
- Gradually the form and meaning of **algorism** became corrupted and resulted into **algorithm**

Properties of algorithm

Properties of algorithm

- **Input** — an algorithm has zero or more inputs

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminates after finite number of steps

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminate after finite number of steps
- **Definiteness** — each step of algorithm needs to be defined precisely and unambiguously

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminate after finite number of steps
- **Definiteness** — each step of algorithm needs to be defined precisely and unambiguously
 - *add salt to taste*

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminate after finite number of steps
- **Definiteness** — each step of algorithm needs to be defined precisely and unambiguously
 - *add salt to taste*
- **Effectiveness** — operations must all be sufficiently basic that they can in principle be done exactly and in a finite length of time by someone using pencil and paper

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminate after finite number of steps
- **Definiteness** — each step of algorithm needs to be defined precisely and unambiguously
 - *add salt to taste*
- **Effectiveness** — operations must all be sufficiently basic that they can in principle be done exactly and in a finite length of time by someone using pencil and paper
 - *If 4 is the largest integer n for which there is a solution to the equation $w^n + x^n + y^n = z^n$ in positive integers w, x, y , and z , then go to step 6*

Properties of algorithm

- **Input** — an algorithm has zero or more inputs
- **Output** — an algorithm has one or more outputs
- **Finiteness** — an algorithm must terminate after finite number of steps
- **Definiteness** — each step of algorithm needs to be defined precisely and unambiguously
 - *add salt to taste*
- **Effectiveness** — operations must all be sufficiently basic that they can in principle be done exactly and in a finite length of time by someone using pencil and paper
 - *If 4 is the largest integer n for which there is a solution to the equation $w^n + x^n + y^n = z^n$ in positive integers w, x, y , and z , then go to step 6*
- *An algorithm is an explicit, precise, unambiguous, mechanically-executable sequence of elementary instructions, usually intended to accomplish a specific purpose. — Jeff Erickson*

Overview

- Algorithm and program
- Pseudo-code
- Algorithm + Data-Structures = Program
- Initial solution + Analysis + Solution Refinement + Data-Structures = Final Program
- Use of recursive definition for initial solution
- Use recurrence equation for proofs and analysis
- Solution refinement through recursion transformation and traversal
- Data structures for saving past results for future use
- Sample problems
 - Finding MAX
 - Finding MAX and MIN
 - Finding MAX and 2nd-MAX
 - Fibonacci numbers
 - Searching in ordered / unordered list
 - Sorting
 - Pattern matching
 - Permutation and combination
 - Shortest path

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:
 1. $\max(L)$
 2. if $|L| = 1$ return x_1
 3. $L' = L - \{x_1\}$
 4. $x' = \max(L')$
 5. if $(x_1 > x')$ return x_1
 6. else return x'

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

$\{5, 6, 10, 3, 1, 12, 2\}$

1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

1. $\max(L)$

2. if $|L| = 1$ return x_1

3. $L' = L - \{x_1\}$

4. $x' = \max(L')$

5. if $(x_1 > x')$ return x_1

6. else return x'

$\{5, 6, 10, 3, 1, 12, 2\}$
↙
 $x_1 = 5, L' = \{6, 10, 3, 1, 12, 2\}$

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

1. $\max(L)$

2. if $|L| = 1$ return x_1

3. $L' = L - \{x_1\}$

4. $x' = \max(L')$

5. if $(x_1 > x')$ return x_1

6. else return x'

$\{5, 6, 10, 3, 1, 12, 2\}$

$\swarrow$
 $x_1 = 5, L' = \{6, 10, 3, 1, 12, 2\}$

$\searrow$
 $x_1 = 6, L' = \{10, 3, 1, 12, 2\}$

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

1. $\max(L)$

2. if $|L| = 1$ return x_1

3. $L' = L - \{x_1\}$

4. $x' = \max(L')$

5. if $(x_1 > x')$ return x_1

6. else return x'

$\{5, 6, 10, 3, 1, 12, 2\}$

$x_1 = 5, L' = \{6, 10, 3, 1, 12, 2\}$

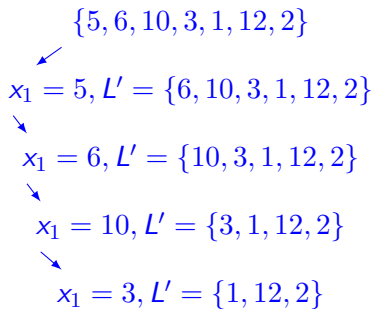
$x_1 = 6, L' = \{10, 3, 1, 12, 2\}$

$x_1 = 10, L' = \{3, 1, 12, 2\}$

Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

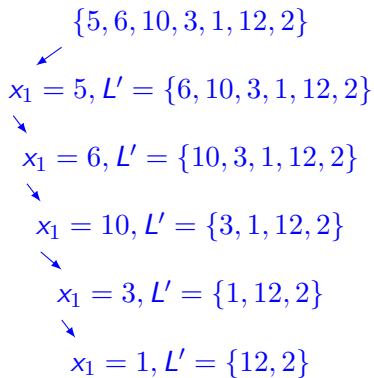
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

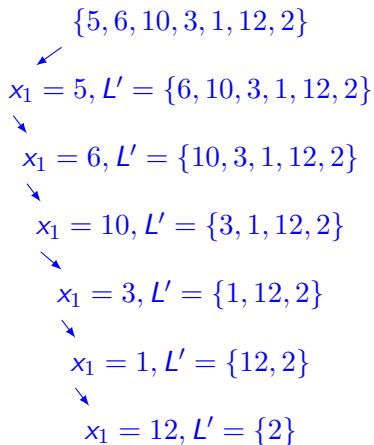
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

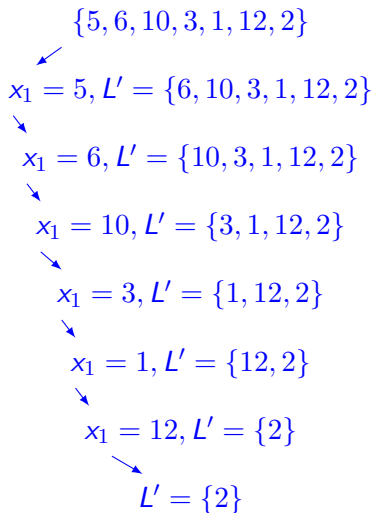
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

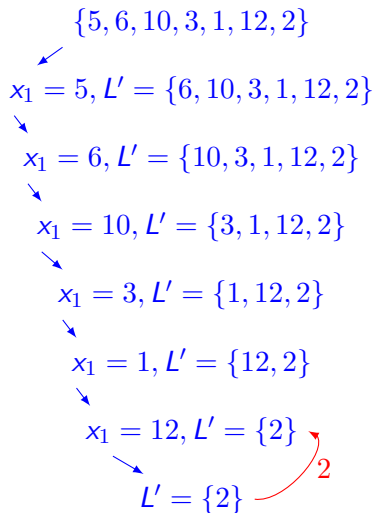
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

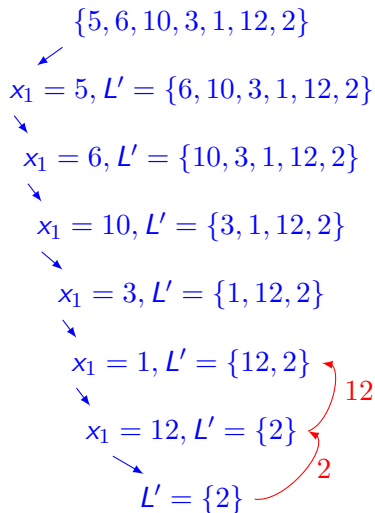
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

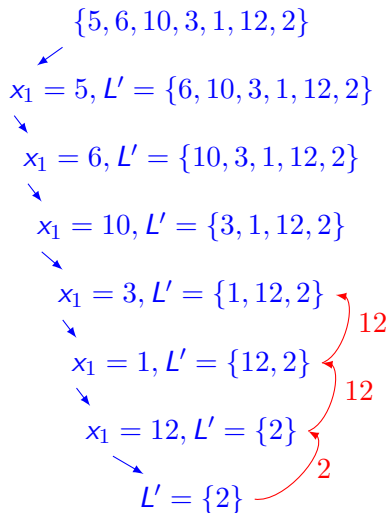
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

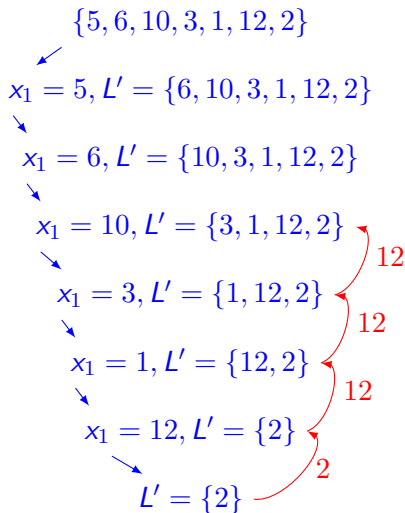
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

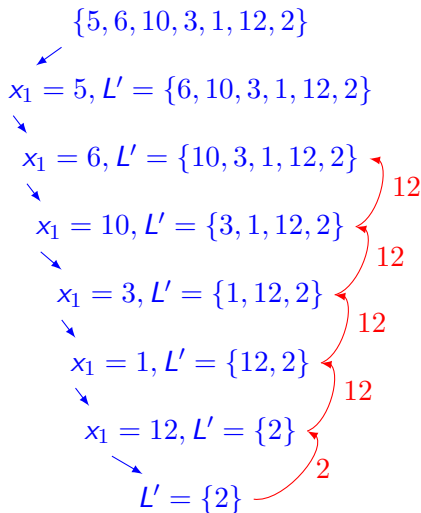
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

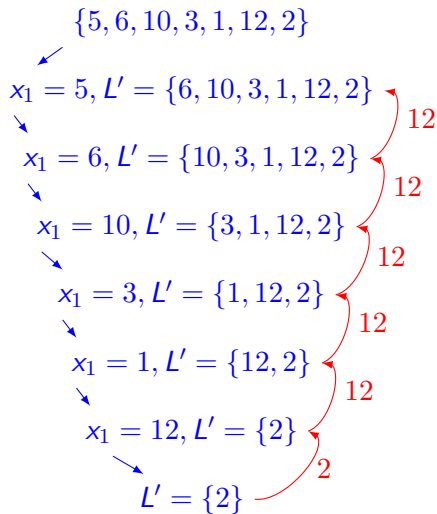
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

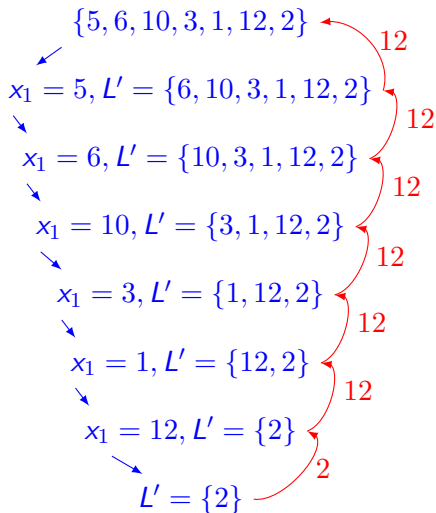
1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'



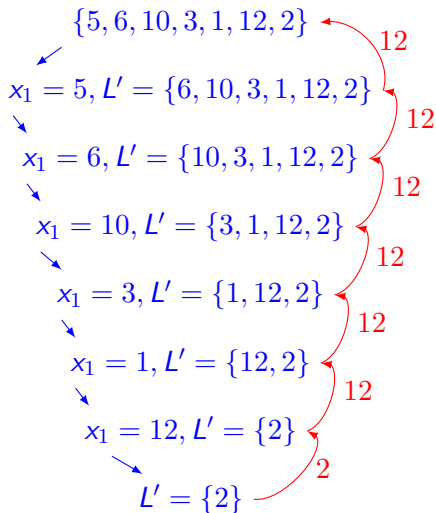
Finding MAX of n elements (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Sequential comparison:

1. $\max(L)$
2. if $|L| = 1$ return x_1
3. $L' = L - \{x_1\}$
4. $x' = \max(L')$
5. if $(x_1 > x')$ return x_1
6. else return x'

Complexity analysis:

$$\begin{aligned} T(n) &= T(n-1) + 1, & n > 1 \\ &= 0, & n = 1 \end{aligned}$$



Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Recursive formulation:
 1. $\text{max2}(L)$
 2. if $|L| = 1$ return x_1
 3. Split L into 2 non-empty sets L_1, L_2
 4. $x = \text{max2}(L_1)$
 5. $y = \text{max2}(L_2)$
 6. if $(x > y)$ return x
 7. else return y

Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\text{max2}(L)$

2. if $|L| = 1$ return x_1

3. Split L into 2 non-empty sets L_1, L_2

4. $x = \text{max2}(L_1)$

5. $y = \text{max2}(L_2)$

6. if $(x > y)$ return x

7. else return y

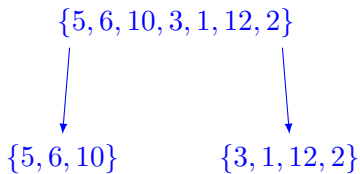
$\{5, 6, 10, 3, 1, 12, 2\}$

Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

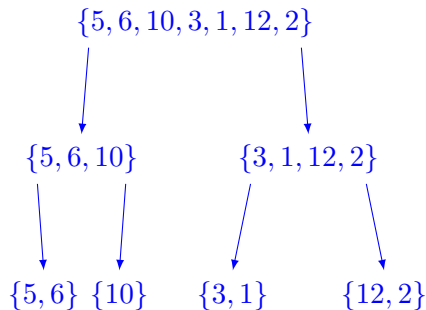


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

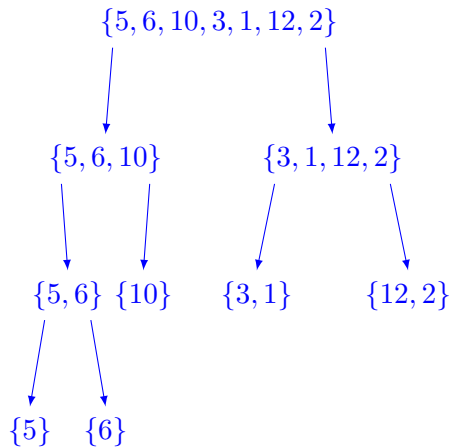


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

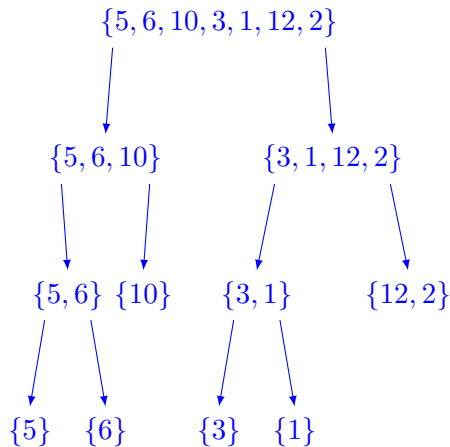


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

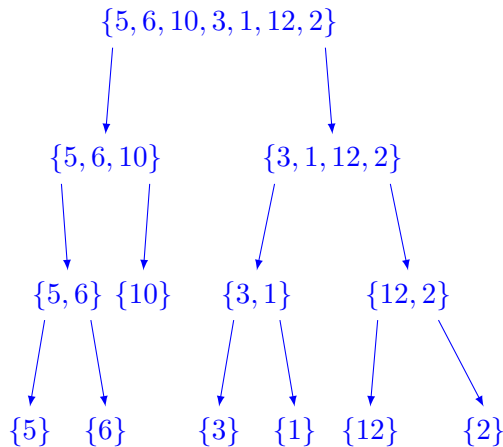


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

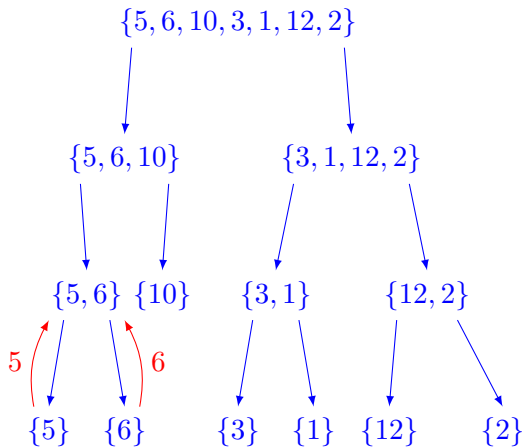


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

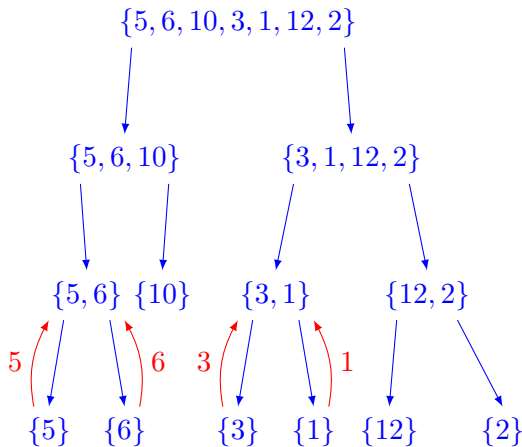


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

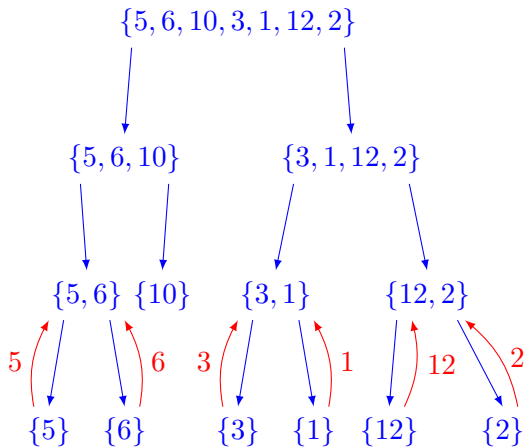


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

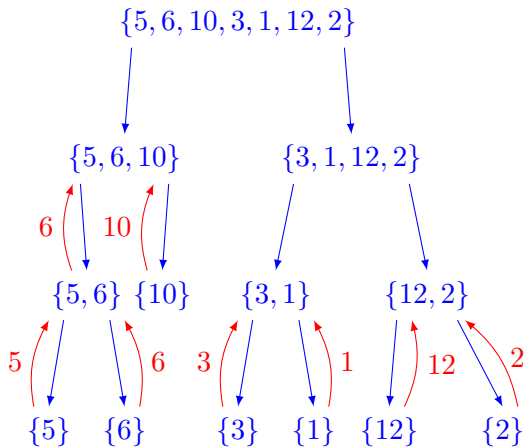


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

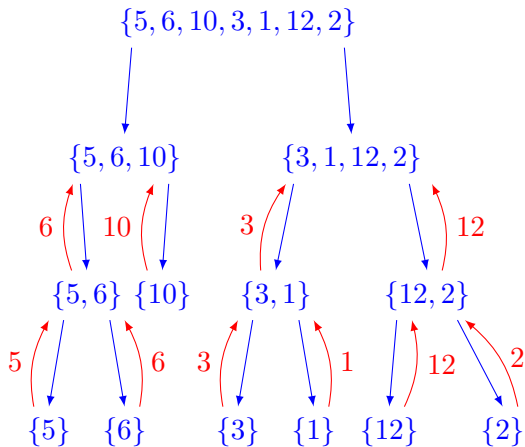


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

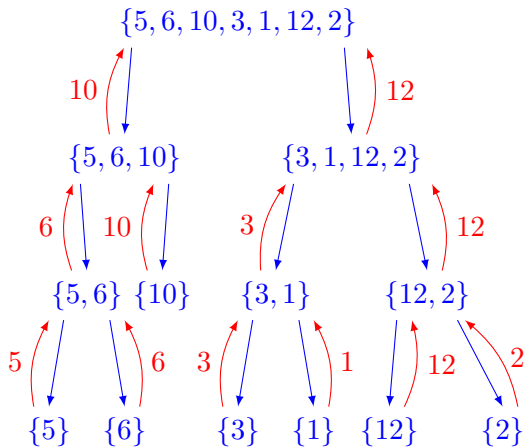


Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

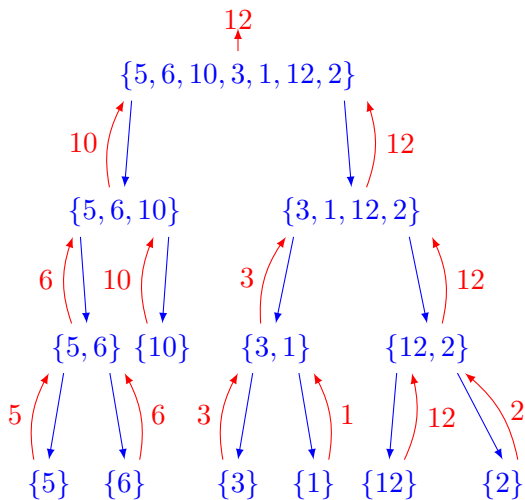
1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y



Finding MAX of n elements (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$
- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y



Finding MAX of n elements (2)

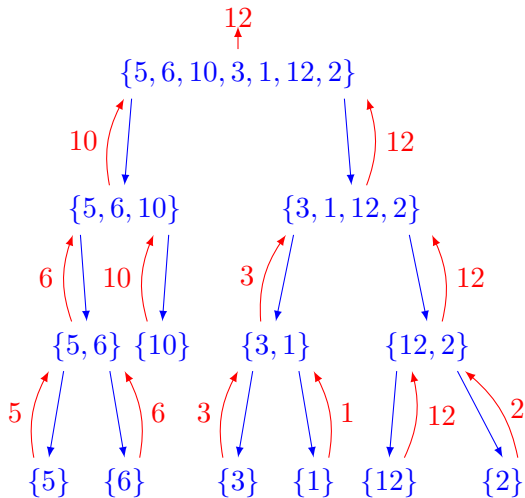
- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$

- Recursive formulation:

1. $\max2(L)$
2. if $|L| = 1$ return x_1
3. Split L into 2 non-empty sets L_1, L_2
4. $x = \max2(L_1)$
5. $y = \max2(L_2)$
6. if $(x > y)$ return x
7. else return y

Complexity analysis:

$$\begin{aligned} T(n) &= T(n-k) + T(k) + 1, & n > 1 \\ &= 0, & n = 1 \end{aligned}$$

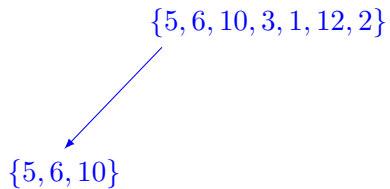


Recursion Flow: Finding MAX of n elements (2)

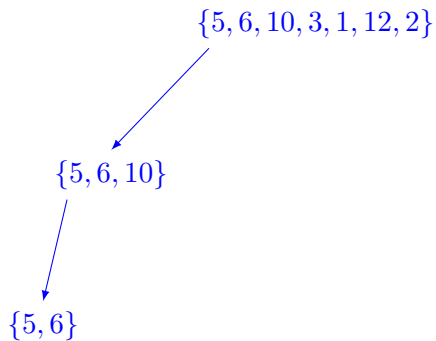
Recursion Flow: Finding MAX of n elements (2)

{5, 6, 10, 3, 1, 12, 2}

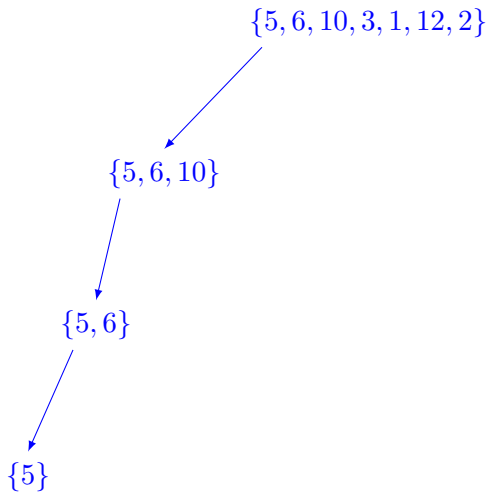
Recursion Flow: Finding MAX of n elements (2)



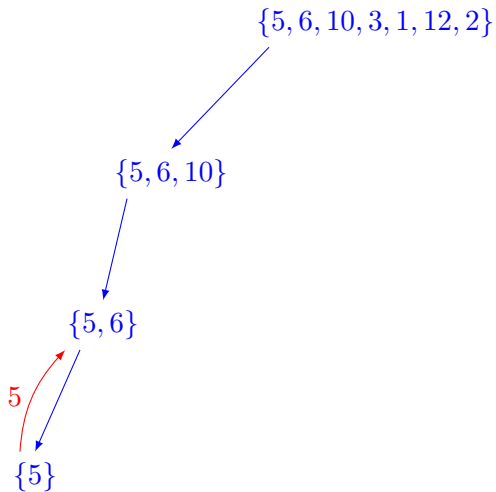
Recursion Flow: Finding MAX of n elements (2)



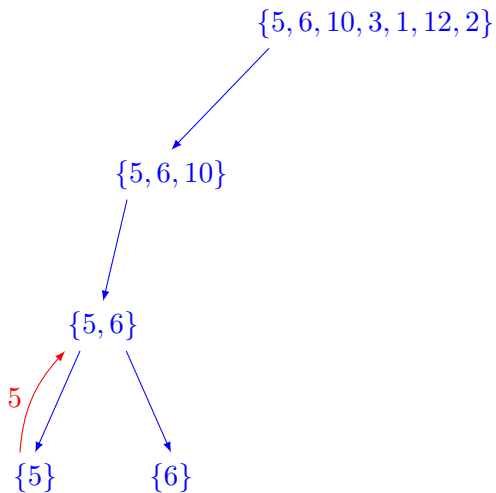
Recursion Flow: Finding MAX of n elements (2)



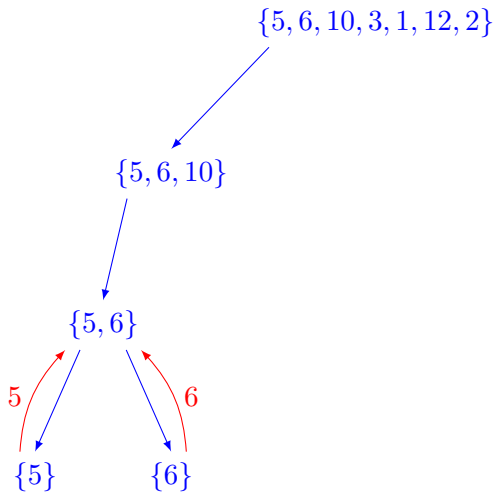
Recursion Flow: Finding MAX of n elements (2)



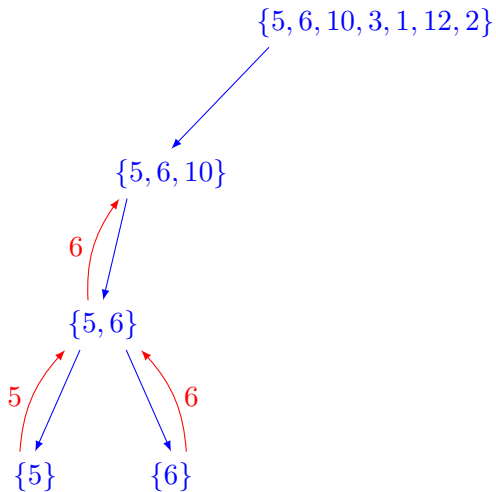
Recursion Flow: Finding MAX of n elements (2)



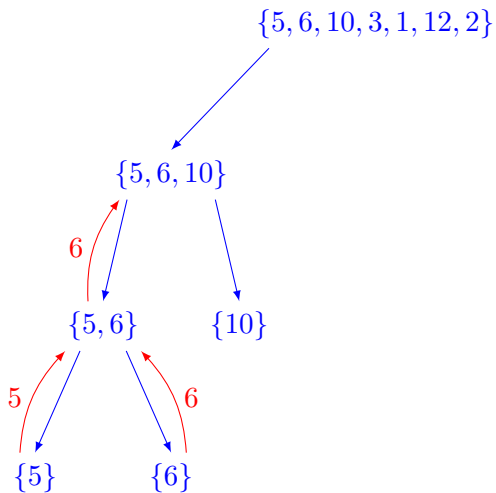
Recursion Flow: Finding MAX of n elements (2)



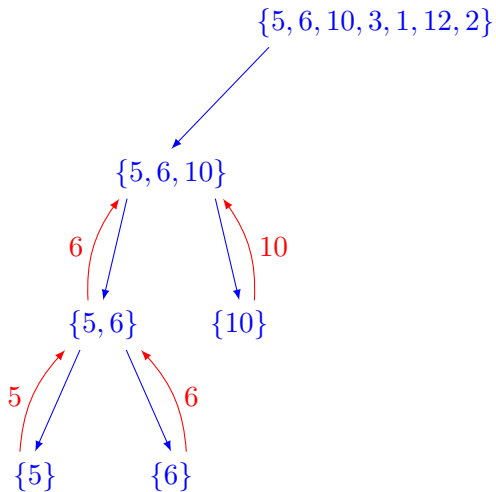
Recursion Flow: Finding MAX of n elements (2)



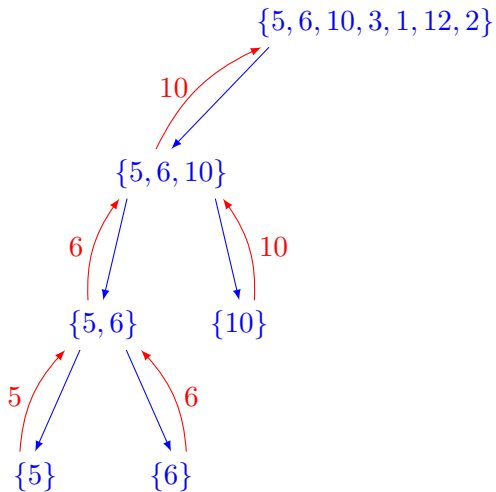
Recursion Flow: Finding MAX of n elements (2)



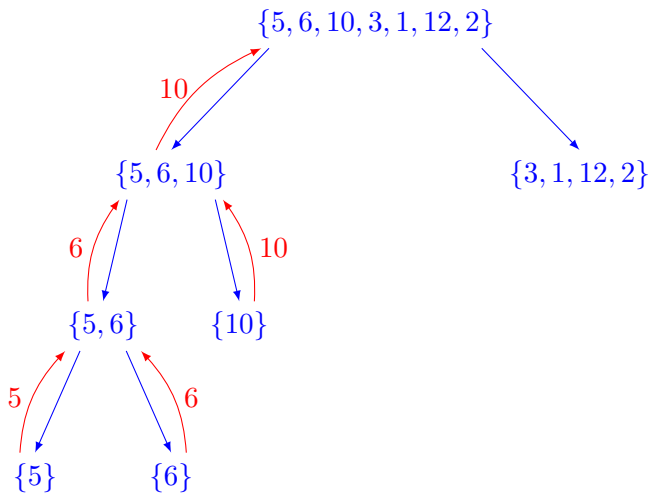
Recursion Flow: Finding MAX of n elements (2)



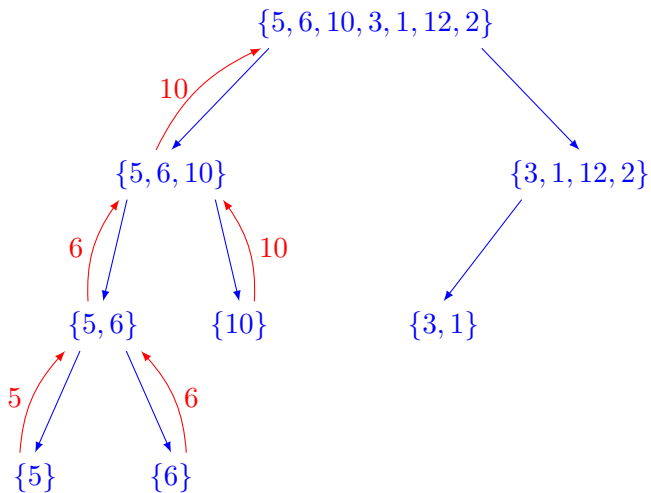
Recursion Flow: Finding MAX of n elements (2)



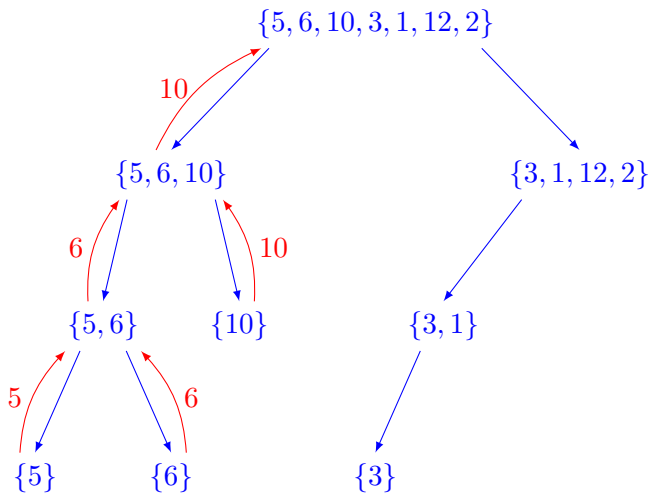
Recursion Flow: Finding MAX of n elements (2)



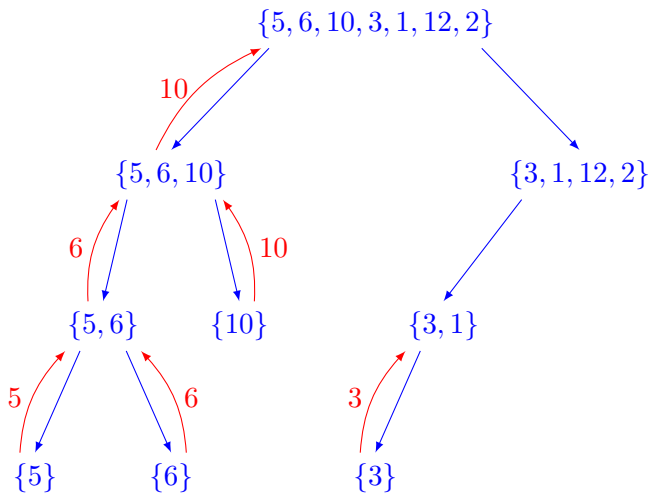
Recursion Flow: Finding MAX of n elements (2)



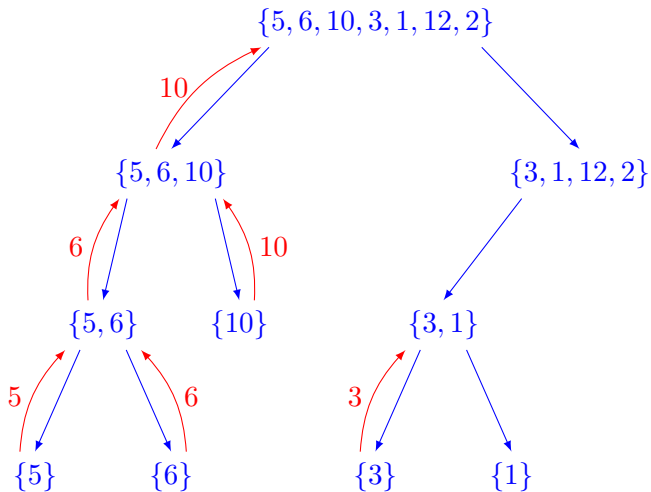
Recursion Flow: Finding MAX of n elements (2)



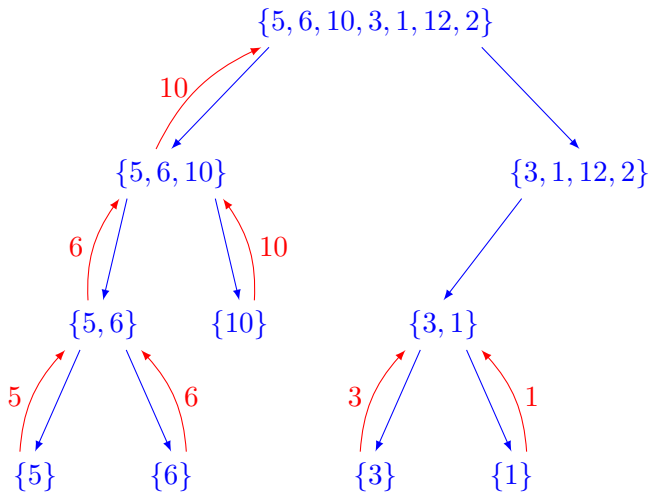
Recursion Flow: Finding MAX of n elements (2)



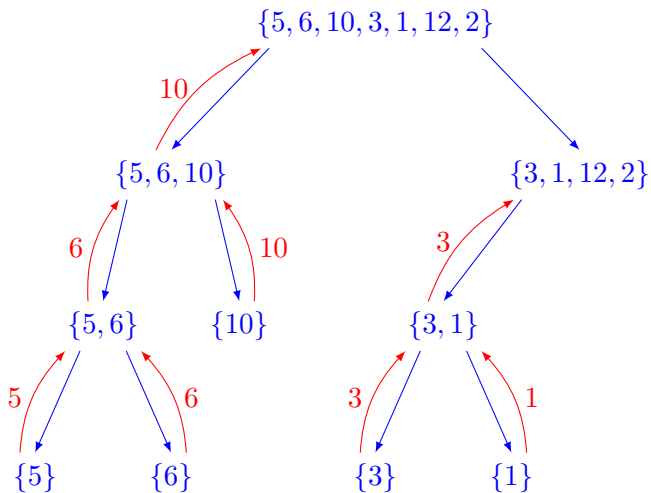
Recursion Flow: Finding MAX of n elements (2)



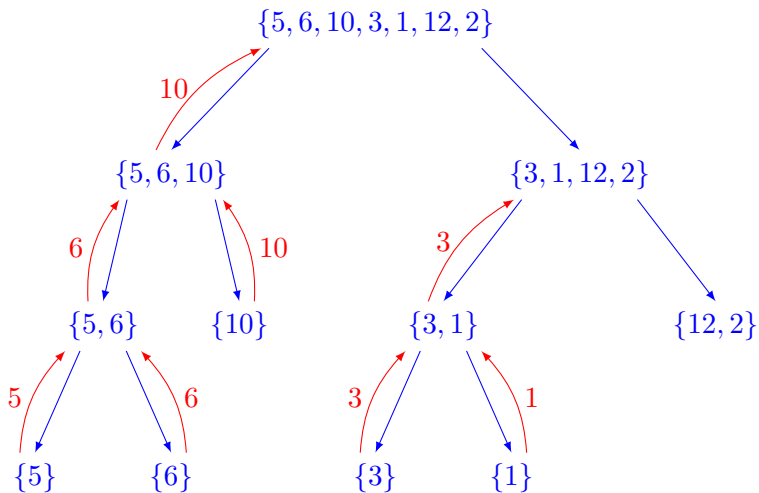
Recursion Flow: Finding MAX of n elements (2)



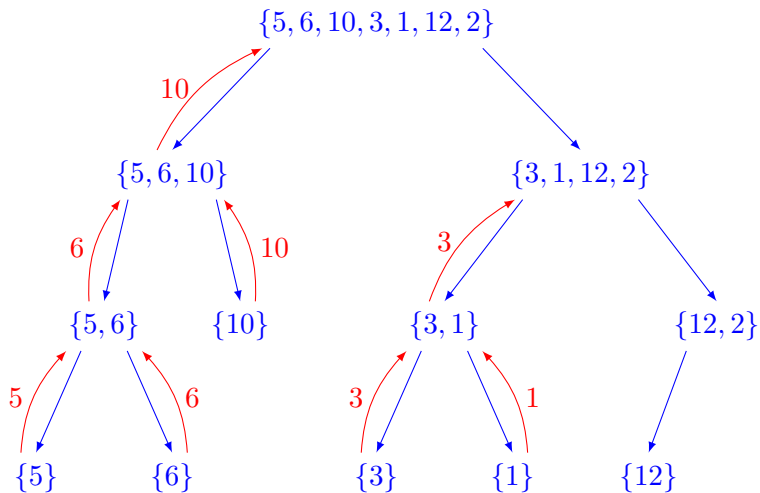
Recursion Flow: Finding MAX of n elements (2)



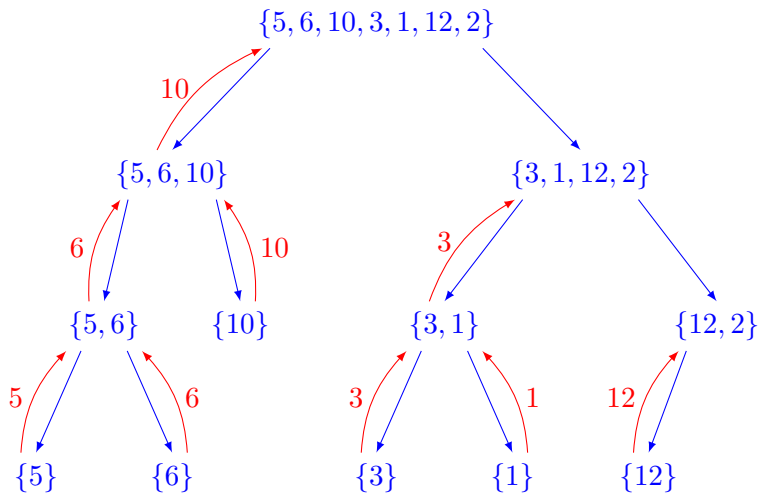
Recursion Flow: Finding MAX of n elements (2)



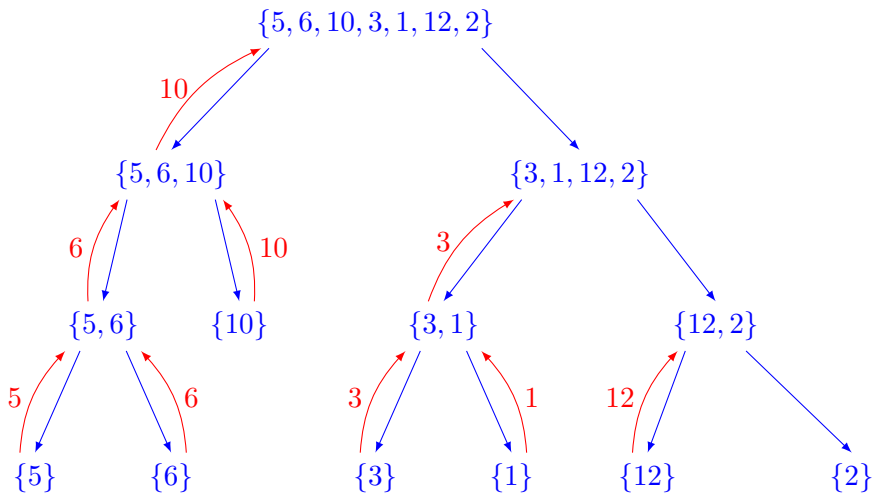
Recursion Flow: Finding MAX of n elements (2)



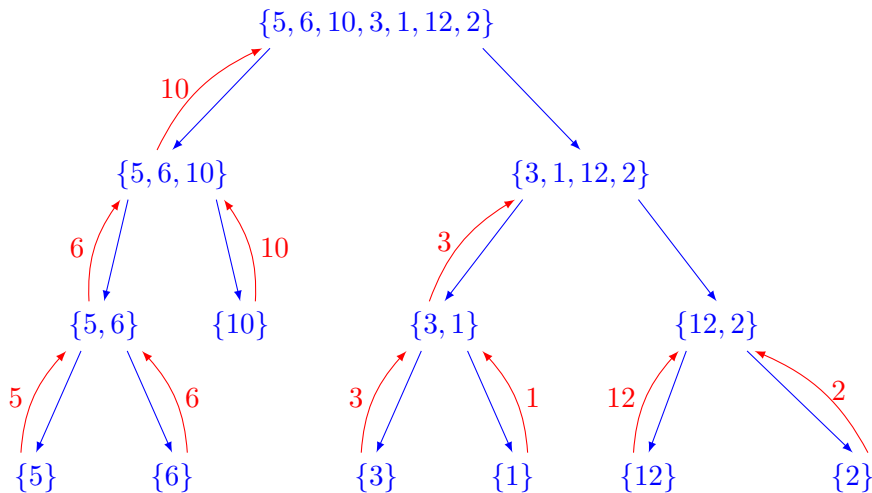
Recursion Flow: Finding MAX of n elements (2)



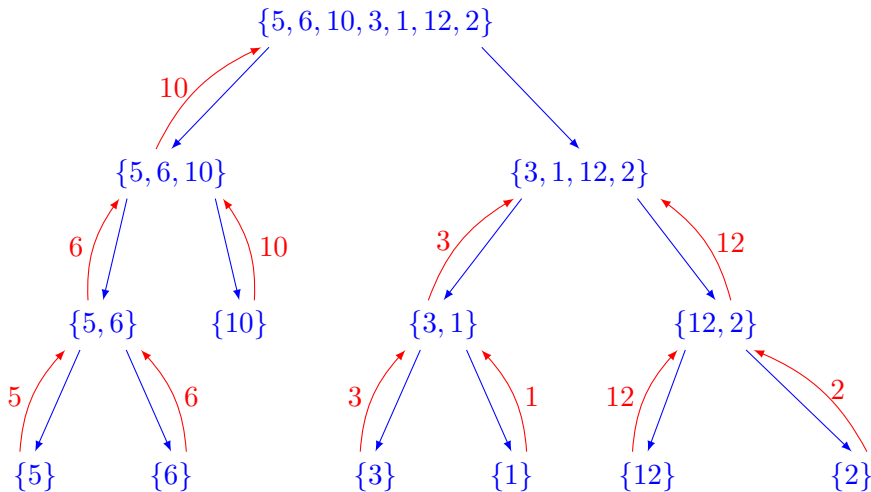
Recursion Flow: Finding MAX of n elements (2)



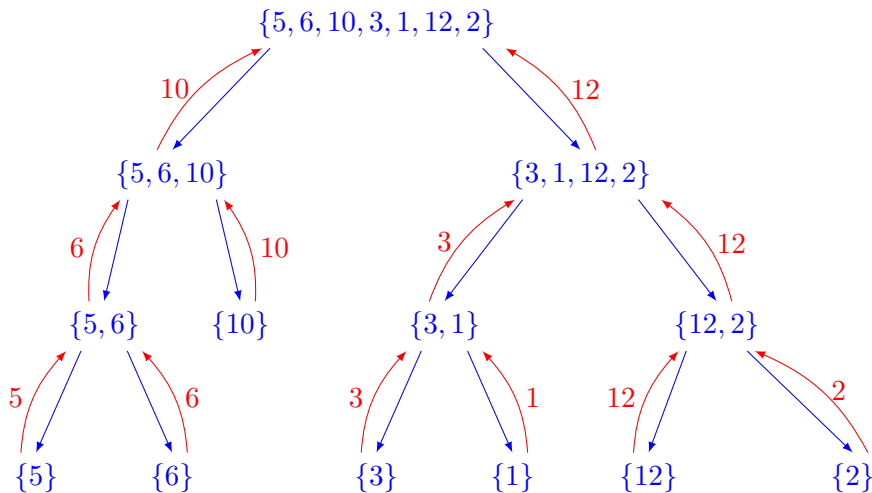
Recursion Flow: Finding MAX of n elements (2)



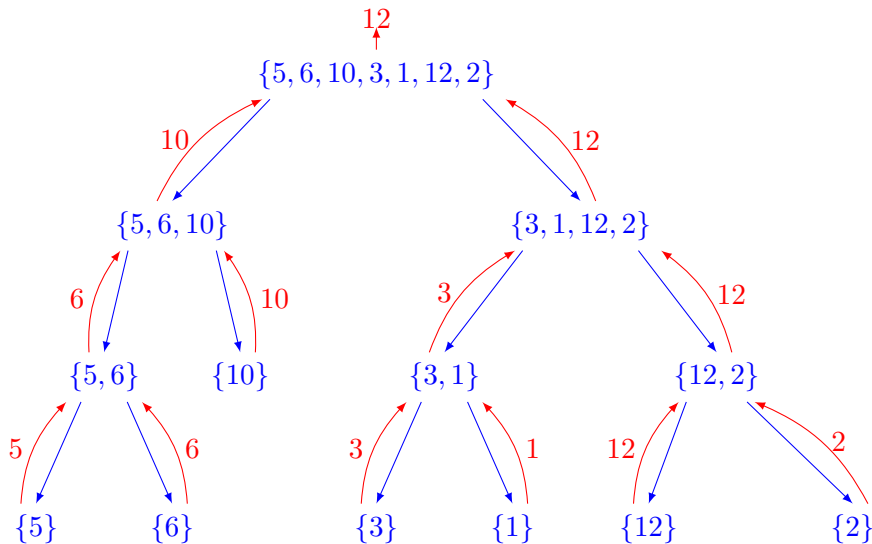
Recursion Flow: Finding MAX of n elements (2)



Recursion Flow: Finding MAX of n elements (2)



Recursion Flow: Finding MAX of n elements (2)



Comparison Tournament

- Finding of maximum can be viewed as a tournament of players taken two at a time

x_0

x_1

x_2

x_3

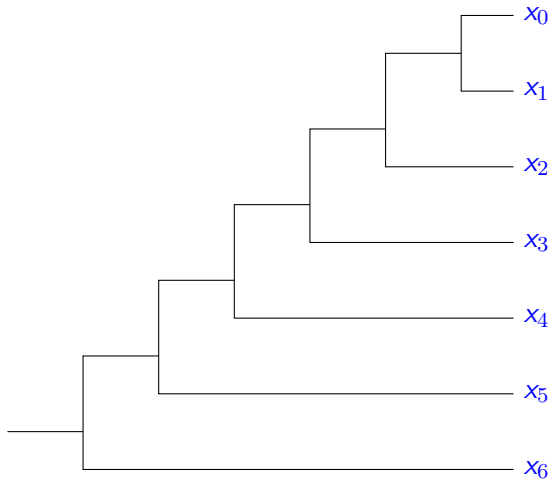
x_4

x_5

x_6

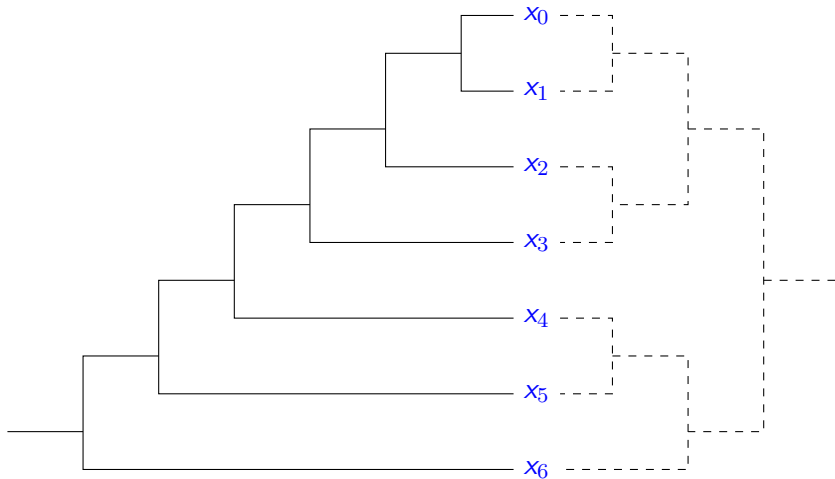
Comparison Tournament

- Finding of maximum can be viewed as a tournament of players taken two at a time



Comparison Tournament

- Finding of maximum can be viewed as a tournament of players taken two at a time



MAX & MIN (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\min\{L\}$

MAX & MIN (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\min\{L\}$
- Sequential comparison
 1. $\text{maxmin}(L)$
 2. if $|L|=1$ return $\langle x_1, x_1 \rangle$
 3. $L' = L - \{x_1\}$
 4. $\langle y_1, y_2 \rangle = \text{maxmin}(L')$
 5. if $x_1 > y_1$ then $m_1 = x_1$ else $m_1 = y_1$
 6. if $x_1 < y_2$ then $m_2 = x_1$ else $m_2 = y_2$
 7. return $\langle m_1, m_2 \rangle$

MAX & MIN (2)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\min\{L\}$
- Recursive definition
 1. $\text{maxmin2}(L)$
 2. if $|L|=1$ return $\langle x_1, x_1 \rangle$
 3. if $|L|=2$ if $x_1 > x_2$ return $\langle x_1, x_2 \rangle$ else return $\langle x_2, x_1 \rangle$
 4. Split L into 2 non-empty sets L_1, L_2
 5. $\langle y_1, y_2 \rangle = \text{maxmin2}(L_1)$
 6. $\langle z_1, z_2 \rangle = \text{maxmin2}(L_2)$
 7. if $y_1 > z_1$ then $m_1 = y_1$ else $m_1 = z_1$
 8. if $y_2 < z_2$ then $m_2 = y_2$ else $m_2 = z_2$
 9. return $\langle m_1, m_2 \rangle$

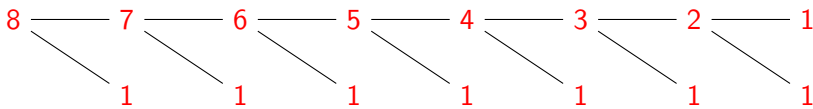
MAX & MIN (3)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\min\{L\}$
- Recursive definition - Choice of split
 - Recurrence relation:

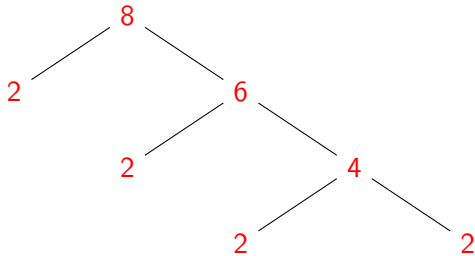
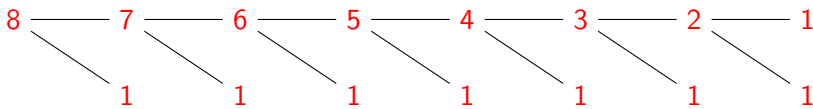
$$\begin{aligned}T(n) &= 0, & n = 1 \\&= 1, & n = 2 \\&= T(k) + T(n - k) + 2, & n = 2\end{aligned}$$

Options for splitting

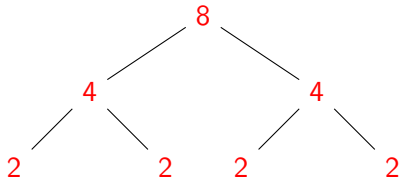
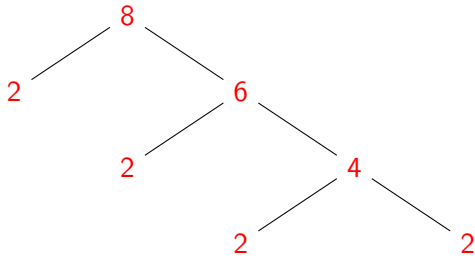
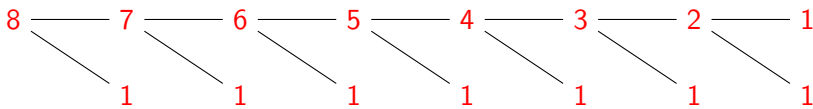
Options for splitting



Options for splitting



Options for splitting



MAX & 2nd-MAX (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\text{2nd-max}\{L\}$

MAX & 2nd-MAX (1)

- Given $L = \{x_1, x_2, \dots, x_n\}$, all x_i are integers. We need to find $\max\{L\}$ and $\text{2nd-max}\{L\}$
- Recursive definition
 - $\text{max2ndmax}(L)$
 - if $|L|=1$ return $\langle x_1, x_1 \rangle$
 - if $|L|=2$ if $x_1 > x_2$ return $\langle x_1, x_2 \rangle$ else return $\langle x_2, x_1 \rangle$
 - Split L into 2 non-empty sets L_1, L_2
 - $\langle y_1, y_2 \rangle = \text{max2ndmax}(L_1)$
 - $\langle z_1, z_2 \rangle = \text{max2ndmax}(L_2)$
 - if $(y_1 > z_1)$ { $m_1 = y_1$; $m_2 = z_1 > y_1 ? z_1 : y_1$; }
 - else { $m_1 = z_1$; $m_2 = y_1 > z_2 ? y_1 : z_2$; }
 - return $\langle m_1, m_2 \rangle$
- Explore different splitting options

MAX & 2nd-MAX (2)

MAX & 2nd-MAX (2)

$\{5, 6, 10, 3, 1, 12, 2\}$

MAX & 2nd-MAX (2)

{5, 6, 10, 3, 1, 12, 2}

1	2	3	4	5	6	7	8	9	10	11	12	13

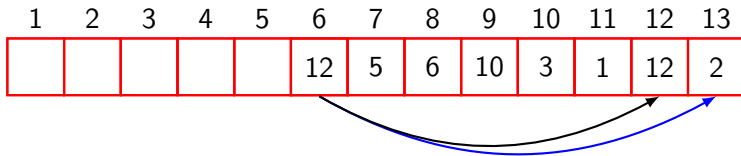
MAX & 2nd-MAX (2)

{5, 6, 10, 3, 1, 12, 2}

1	2	3	4	5	6	7	8	9	10	11	12	13
						5	6	10	3	1	12	2

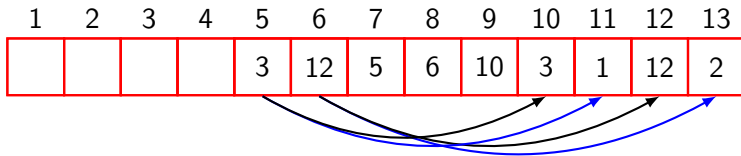
MAX & 2nd-MAX (2)

$\{5, 6, 10, 3, 1, 12, 2\}$



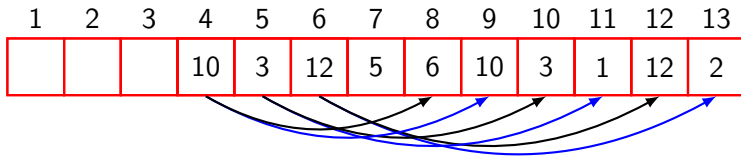
MAX & 2nd-MAX (2)

{5, 6, 10, 3, 1, 12, 2}



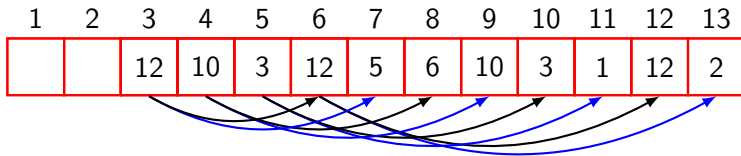
MAX & 2nd-MAX (2)

{5, 6, 10, 3, 1, 12, 2}



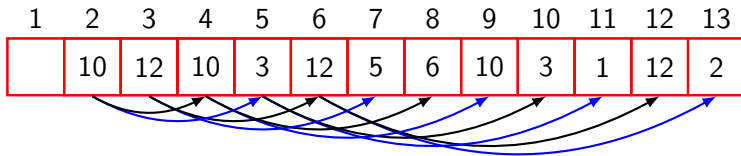
MAX & 2nd-MAX (2)

{5, 6, 10, 3, 1, 12, 2}



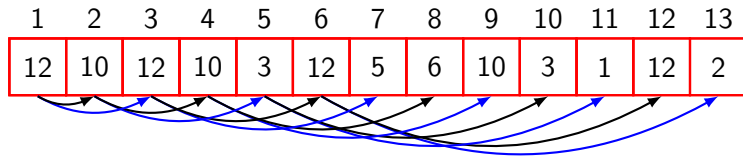
MAX & 2nd-MAX (2)

$\{5, 6, 10, 3, 1, 12, 2\}$



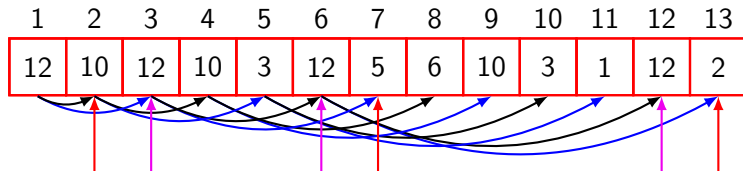
MAX & 2nd-MAX (2)

$\{5, 6, 10, 3, 1, 12, 2\}$



MAX & 2nd-MAX (2)

$\{5, 6, 10, 3, 1, 12, 2\}$



Exercise!

1. $\max(L = \{x_1, \dots, x_n\})$
2. $m = x_1$
3. for $i = 2$ to n
4. if $m < x_i$
5. $m = x_i$
6. return m

- How many times will line 5 be executed, on average?

Thank you!